Learning theories for noun-phrase sentiment composition

Petrakis, S ; Klenner, M

Abstract: The work presented here is an approach to Sentiment Analy- sis from a rule-based, compositional perspective. The proposed approach is characterized by three major points: (a) rules are automatically learned from annotated corpora using Inductive Logic Programming and represented as Prolog sets of clauses, (b) the focus is on the noun-phrase (NP) level, and (c) learning is performed on deep-parsed structures. We describe the process of annotating a collection of some 3000 German NPs of medium to quite complex structure, as well as the empirical evaluation of our implementation, in comparison with commonly used classifiers and a handcrafted rule-based system.

Originally published at:
Petrakis, S; Klenner, M (2011). Learning theories for noun-phrase sentiment composition. In: 8th International NLPCS Workshop, Kopenhagen, 20 August 2011 - 21 August 2011. Samfundslitteratur, 179-188.

Learning Theories for Noun-Phrase Sentiment Composition

Stefanos Petrakis and Manfred Klenner

Institute for Computational Linguistics, University of Zurich, Zurich, Switzerland,
{petrakis,klenner}@cl.uzh.ch,
http://www.cl.uzh.ch/

Abstract. The work presented here is an approach to Sentiment Analysis from a rule-based, compositional perspective. The proposed approach is characterized by three major points: (a) rules are automatically learned from annotated corpora using Inductive Logic Programming and represented as Prolog sets of clauses, (b) the focus is on the noun-phrase (NP) level, and (c) learning is performed on deep-parsed structures. We describe the process of annotating a collection of some 3000 German NPs of medium to quite complex structure, as well as the empirical evaluation of our implementation, in comparison with commonly used classifiers and a handcrafted rule-based system.

Keywords: sentiment analysis, inductive logic programming, machine learning

1 Introduction

Sentiment detection is a young field of research that has been progressing rapidly into maturity during recent years. During its development, the term itself, sentiment detection (or analysis[1]), has conceptually annexed neighboring research fields, namely subjectivity analysis and opinion mining [?]. The work presented here focuses on "pure" sentiment detection, striving to correctly evaluate the polarity orientation (e.g. positive/negative/neutral) of sentiment in text.

The majority of approaches to sentiment detection that have been proposed in the past are (a) focusing on the document level and (b) employing standard statistical machine learning techniques to perform classification of polarity[2]. In contrast to these approaches, a sub-current of research within the field has focused on explicit compositional treatment of sentiment on the sentential as well as the subsentential levels by utilizing more of the available linguistic structures [?], [?], [?]. The compositional nature of sentiment is clearly manifested in common phenomena like negation ('these were not good news'), intensification ('an excellent liar') and diminishment ('an unrealistic hope') among others.
The work presented here embraces this compositional view of sentiment analysis.

[1] In the rest of the paper we use these two terms interchangeably.
[2] [?] provide a detailed list of such approaches.

We describe a method for automatically producing theories of sentiment composition for noun-phrases(NPs). This is a supervised machine learning approach which intends to model relations between the constituents of NPs. From such a model, sentiment on the NP level could then be systematically calculated. While the language of choice for the specific experiment is German, the idea is applicable to any given language. Learning is performed on relationally structured data, a corpus built specifically for this purpose.
In the rest of this paper we shall describe the corpus annotation process as well as the relational learning component our system employs. We compared an implementation of our proposed approach with standard statistical sentiment classifiers as well as a handcrafted rule-based system and will present the results of the empirical evaluation. We will conclude with insights gained and remarks regarding the applicability and significance of this novel idea.

2 Sentiment Composition

The compositional aspect of sentiment can be expressed by the following statement, a slightly modified version of the principle of compositionality[**?**]:

> *"The sentiment of a complex expression is determined by its structure and the sentiment of its constituents"*

Correctly analyzing the sentiment of a chunk, phrase, sentence or clause can be accomplished by identifying and utilizing the relations between the constituents of the given textual unit. A number of basic examples of such relations on the NP level are given in [**?**]:

ADJ	NOUN	→ NP	Example
NEG	POS	→ NEG	disappointed hope
NEG	NEG	→ NEG	a horrible liar
POS	POS	→ POS	a good friend
POS	NEG	→ NEG	a perfect misery
POS	NEU	→ POS	a perfect meal
NEG	NEU	→ NEG	a horrible meal

Fig. 1. NP Sentiment Composition

The patterns for sentiment composition of NPs observed in Figure **??** are quite intuitive and one could easily formulate them as rules. For example, the case of "disappointed hope" in the first line is one of many instatiations of the general pattern where a NP comprises a negative (NEG) adjective (ADJ) and a positive noun. However, manually producing such rules for NP sentiment composition is not always going to be a trivial task since NPs do not always come in such simple form.
Consider the following example of a NP:

> ”...a case of less than extreme violence that was not an expression of aggression, only the reaction to a stressful and fear-inducing situation...”

The interactions between smaller parts of this example like ”not” and ”expression of aggression” or even bigger ones like ”was not an expression of aggression” and ”only a reaction...” are essential factors for the overall sentiment that springs out of this complex NP. Expressing the set of all these interactions as a single rule that would capture this specific and also similar cases is a challenging task. There are many more examples that can illustrate how the task of manually producing rules for NP sentiment composition can be mentally-intensive as well as time-demanding.

2.1 Theory vs. Praxis

Identifying relations between words, chunks, phrases or sentences, whether labeled as interactions or dependencies, allows us to model sentiment on a high dimensional space. Whether this is necessary or not, is an open question, one that we tried to answer empirically by measuring the performance of such an implementation.
However, what is of vital importance for our research is the actual theory at hand. We are interested in research that can be applied in the real world, but what we consider as our priority in this specific research attempt is to produce a concise and highly detailed theory of compositional sentiment analysis.

2.2 Focus on the Phrase Level

There are two primary incentives why one would go for sentiment detection on the phrase level:

- There is an actual need for analyzing input at this level. Examples of text at the phrase level are found abundantly and frequently on the web in the form of statusa, tweets and other similar, fragmented, stand-alone units of texts.
- For an integrated system that performs sentiment detection in a bottom-up direction, correctly detecting sentiment on the sentence and higher levels means above all correctly detecting sentiment at the lower levels. We intend to construct such a system in the future which is why having a reliable compositional component for the phrase level is an important requirement.

2.3 Related Work

We consider as related to our work approaches that are either specifically focusing on the phrasal sentiment or are able to handle such input and at the same time accomplish their goal in a entirely or partly compositional way.

The extensive theoretical framework proposed by [?] is manually devised in contrast to ours. The authors also mention that their implementation is based on a non-robust parser which introduces a significant number of errors in the system. As we will describe later, our system is using a robust dependency parser [?].

In [?] polarity classification is based on a number of features that make use of inter-constituent relations. These features are hard-coded and in no case exhaustive. Additionally, their system performs as a first step subjectivity detection (polar/neutral) which we do not consider a crucial step.

The system described in [?] focuses on the sentential and subsentential level and uses among others a set of manually written rules for inferring sentiment in a compositional manner. These rules, in comparison to our approach, apply only to flat structures, i.e. shallow parsed phrases, although the authors hint that deep-parsing could improve their system's performance.

3 Resources

To produce the annotated data needed for learning theories of sentiment composition we: (a) collected a sufficiently large number of NPs, (b) automatically parsed and tagged words with dependency and part-of-speech (POS) information, (c) annotated words with (prior) polarity, (d) selected the most complex and interesting NPs out of the set and (e) manually labeled these with NP polarity information. In the following paragraphs this process is described in more detail.

3.1 Polarity Lexicon

The lexicon we used is based on the manually curated polarity lexicon for German used by the PolArt system [?]. That lexicon has been built using the lexical database GermaNet as reference and contains more than 8,000 words (nouns, verbs and adjectives), which makes it - as of this writing - the most extensive manually produced polarity lexicon for German. Note that the version we used is a modified one as it contains an automatically extended list of adjectives[3] as described in [?].

3.2 Collecting NPs

We decided to use the DeWaC corpus [?] as raw material since it provides a practically unlimited - for our purposes - number of NPs. We parsed and tagged a huge number of sentences from DeWaC using the Pro3Gres parser for German [?]. After that, we extracted only NPs from that set of deep-parsed sentences.
These resulting NPs were now containing dependency and POS information and

[3] A necessary extension, as it was found during evaluation that the number of adjectives it originally contained was quite restricted.

we also added polarity information from our polarity lexicon for German. At this point, we had at hand a huge pool of NPs that were annotated with the information we were interested in for the learning phase. But, in order to learn interesting and sufficiently complex rules from the training material two more post-requirements were introduced; we decided we needed NPs that were (a) as long as possible and (b) as complex and rich in polarity as possible.

The first requirement (a) was straightforward to use as a filtering criterion. The second requirement (b) was further formulated as a set of three sub-criteria, picking NPs that contained:

- a SHIfter and one or more polarized word(s)
- an INTensifier and one or more polarized word(s)
- a NEGative and a POSitive word

where a SHIfter is a word (from the polarity lexicon) that can invert the polarity of another word (as in "*not* good") and an INTensifier is a word that can enhance or reduce the polarity of another word (as in "*excellent* liar").

We applied a selective search based on requirements (a) and (b) inside our pool of NPs and finally extracted 4200 NPs ranging from small to greater complexity and containing few to many polarized words.

3.3 Guiding the Annotation

These 4200 NPs were given to a group of annotators, - two annotators assigned per phrase - that were instructed to annotate them for polarity (positive/negative/neutral). The greatest of the challenges was to guide the annotators to adopt a common point of evaluating sentiment. For given phrases, adopting a subjective view of the sentiment captured by the phrase can easily lead to a variety of evaluations. We wanted to avoid such situations and where possible 'enforce' a uniform evaluation and annotation approach.
To help us and the annotators in this we introduced the notion of "political/common-sense correctness" as a means to keep things under control. A simple but good example of where this criterion could prove useful is:

"...eating cold pizza in the morning..."[4]

While the notion of eating cold food is in principle negative, we should not neglect the fact that (a) some dishes are meant to be served cold and, most importantly, (b) some of us simply enjoy a slice of yesterday's pizza in the morning. However, the criterion of "political/common-sense correctness" should stop an annotator from marking such a phrase as positive and in this way maintain some level of uniform way of evaluating sentiment.

The second point that we explicitely demanded from the annotators was to operate in a "context-independent" way. That meant that they should always

[4] Objections to whether there is sentiment at all in this phrase are reasonable. We think of such cases sentimentally interesting, in the sense that they can provoke sentiment.

try to annotate without filling in missing context information and focus only on the words and sentiment that was present in the phrase. We believe this also to help avoid varied responses.

4 Learning Theories for Sentiment Composition

The art of manually writing rules for sentiment composition should not be underestimated. It is however undoubtedly tedious at times and quite demanding most of the times, especially as the complexity of the structures one wants to model increases.
It is therefore desirable to be able to construct such theories automatically.

4.1 ILP and Aleph

Learning relational models from structured data via inductive inference is the focus of the subfield of Machine Learning called Inductive Logic Programming (ILP), a long standing paradigm for inferring sets of rules that model relations. We used the tool named Aleph[5] from [**?**], which is based on the following standard ILP idea:

- Given logic programs B(ackground) and E(xamples)
- Find a logic program H(ypothesis), $H \in \mathcal{H}$
- Where $B, H \to E$
- For given P(ositive) examples,
  $\forall e \in P, B \wedge H \models e$
- For given N(egative) examples,
  $\forall e \in N, B \wedge H \not\models e$

4.2 Example of an Induced Rule

Given the phrase

> "Entgrenzung ohne Stabilisierung"[6]

which was labeled as negative by the annotators, (this phrase serves as a positive example $e \in P$ for negative polarity), and had the following parse tree[7] (this information is part of the background knowledge B)

[5] Available from http://www.comlab.ox.ac.uk/activities/machinelearning/Aleph/aleph.
[6] "Debordering without stabilization" (translation from German).
[7] Word polarity annotations are also visible in square brackets.

```
          Entgrenzung [neg]
       _________|_________
      /         |          \
      .      ->pp->       ohne [shi]
      |                 _______|_______
      |                /     |        \
Entgrenzung_NN        .  ->pn-> Stabilisierung [pos]
                      |                 |
                      |                 |
                  ohne_APPR  Stabil~is~ier~ung_NN
```

Aleph induced the following rule[8] (this rule is part of the induced hypothesis H), for negative NP sentiment composition:

```
np_pol_neg(A) :-
    depends_BonA(A,B), has_pol(B,shi),
    has_pol(A,pos).
```

which reads:

a phrase that is headed by a POSitive noun A,
which dominates a SHIfter B,
should be labeled as negative.

The crucial point of interest is that Aleph in specific and ILP in general, allows us to extract rulesets from resources such as the annotated corpus we prepared. The most important features of these rulesets, or theories, are: (a) they are readable by and therefore comprehensible by humans, an expert could edit and enhance such an automatically learned theory, (b) they are complex enough to justify the usefulness of this whole process, in comparison with a manually edited system, (c) they can be manipulated in interesting ways, e.g. by applying global constraints on a learned theory, something that other tools of the trade don't readily provide and (d) they are theoretically intriguing, they describe sentiment composition in a thorough and detailed way.

5 Empirical Evaluation

We used 2809 NPs from the total of 4200, the ones where the annotators agreed on. This set was divided into a training subset of 2100 NPs and a testing subset of 709 ones. The training subset contained 1391 NPs labeled as being of negative polarity, 142 labeled as neutral and 567 labeled as positive. All of these NPs were used as positive examples ($e \in P$), for their respective classes. The same NPs were grouped together in pairs and used as negative examples ($e \in N$) for their opposing classes, e.g. the positive and neutral polarity labeled NPs were used as negative examples for the negative polarity class. Aleph learned various theories[9] and we selected the best performing one (in comparison). This learned

[8] Expressed in Prolog syntax.

[9] Aleph comes with a multitude of configuration parameters, modifying and tuning each one of them can lead to numerous different theories being produced.

theory was a ruleset of 157 Prolog clauses, 78 for negative NP polarity and 77 for positive NP polarity, ranging from short and simple clauses with a few predicates to longer and more complex ones. No rules for neutral NPs were part of this specific induced theory, a fact we attributed to the low number of positive examples (142) for the neutral polarity class.

5.1 Testing

The theory Aleph learned, an actual Prolog program, was tested on the 709 NPs of the testing dataset. The performances reported can be seen in the following table.

	Precision	Recall	F-Score
NegPol	0.914	0.747	0.822
PosPol	0.388	0.614	0.476

We can easily observe that for the positive polarity class (PosPol) the F-Score is relatively low, influenced by the low precision score. The negative polarity class (NegPol) on the other hand shows a quite satisfying performance. In the next section we compare other systems' performances on the same task.

5.2 Comparing

We chose to compare our system with a number of readily available classifiers as provided by the WeKa toolkit[10] (DecisionTable, SVM, BayesNet, NaiveBayes, ADTree, BFTree, J48, RandomTree, etc.). In order to train these classifiers we converted the training set into feature-vector format maintaining the same information available to Aleph.
In addition, we also tried out PolArt [?], a rule-based system that uses a similar polarity lexicon and a manually produced set of rules. The following table contains the results from running PolArt and various classifiers on the testing dataset.

	Precision	Recall	F-Score
PolArt NegPol	0.751	0.885	0.813
PosPol	0.603	0.388	0.472
DecisionTable NegPol	0.807	0.893	0.848
PosPol	0.628	0.457	0.529
BayesNet NegPol	0.783	0.688	0.733
PosPol	0.394	0.516	0.447
ADTree NegPol	0.788	0.904	0.842
PosPol	0.610	0.383	0.471
BFTree NegPol	0.778	0.872	0.822
PosPol	0.531	0.367	0.434
NPCompoILP NegPol	0.914	0.747	0.822
PosPol	0.388	0.614	0.476

[10] Available from http://www.cs.waikato.ac.nz/ml/weka/.

We observe that the results are comparable with the learned theory's performance (NPCompoILP). The DecisionTable classifier is the best performing classifier and outperforms NPCompoILP, even with a small difference. The rest of the classifiers seem to perform as good if not worse than NPCompoILP.

6 Concluding Remarks

We are aware that the idea of automatically learning theories of sentiment composition presented here is more of a proof of concept that a full grown system. We still need to evaluate more exhaustively our system, especially in order to analyze and improve the performance drop observed for the positive class. It is also imperative to test our method on languages and resources that are commonly used by the community. Finally, we intend to extend this method to other types of phrases, namely verb phrases and move our focus to higher levels of text like sentences.

Overall, we consider the approach presented here as an attractive one. It fulfills our fundamental requirement for an explicit compositional treatment of sentiment. Furthermore, it accomplishes that in an automatic way, without compromising the wish for a concise, linguistically-grounded, complex theory for sentiment composition. At the same time, based on the evaluation results we can see that the suggested approach shows practical competence, at least for the case of NPs. It performs on par with standardly used classifiers which increases even more it's appeal.

Acknowledgments

This work is funded by the Swiss National Science Foundation (grant 100015_122546/1).

References

1. Baroni M., Bernardini S., Ferraresi A., Zanchetta E.: The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. Language Resources and Evaluation 43 (3): 209-226 (2009)
2. Moilanen K., Pulman S.: Sentiment composition. Proceedings of the Recent Advances in Natural Language Processing International Conference: 378–382 (2007)
3. Choi Y., Cardie C.: Learning with compositional semantics as structural inference for subsentential sentiment analysis. Proceedings of the Conference on Empirical Methods in Natural Language Processing: 793–801 (2008)
4. Klenner M., Petrakis S., Fahrni A.: Robust compositional polarity classification. Proceedings of the Recent Advances in Natural Language Processing International Conference: 180-184 (2009)
5. Tang H., Tan S., Cheng X.: A survey on sentiment detection of reviews. Expert Systems with Applications 36 (7): 10760-10773 (2009)
6. Pang B., Lee L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2 (1-2): 1-135 (2008)

7. Dowty D.R., Wall R.E., Peters S.: Introduction to Montague semantics (1981)
8. Clematide S., Klenner M.: Evaluation and Extension of a Polarity Lexicon for German. Computational Approaches to Subjectivity and Sentiment Analysis, ECAI: 7-13 (2010)
9. Sennrich R., Schneider G., Volk M., Warin M.: A New Hybrid Dependency Parser for German. Proceedings of the Biennial GSCL Conference: 115-124 (2009)
10. Wilson T., Wiebe J., Hoffmann P.: Recognizing contextual polarity in phrase-level sentiment analysis. Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing: 347-354 (2005)
11. Srinivasan A.: The aleph manual. Computing Laboratory, Oxford University (2005)